

บทที่ 8 กรณีศึกษา

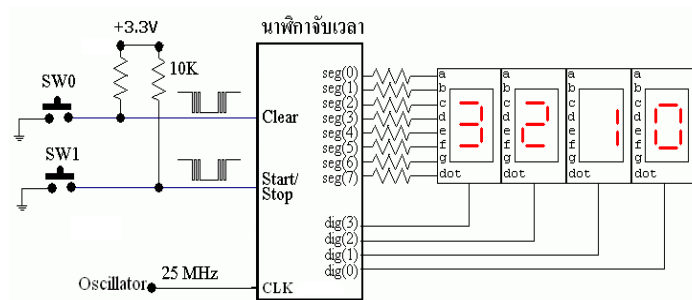
นาฬิกาจับเวลา



08-Case Study

1

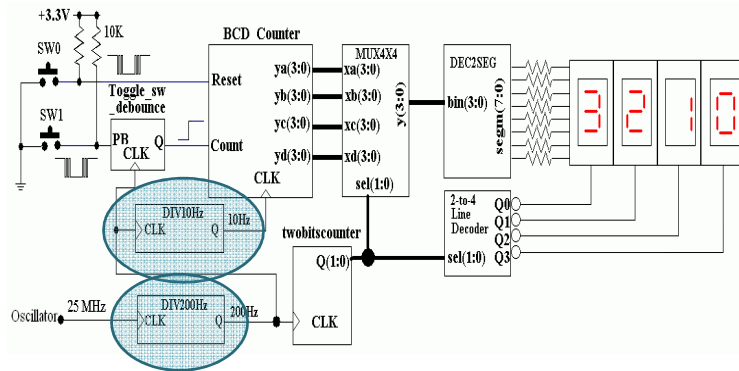
แผนภาพของนาฬิกาจับเวลา



08-Case Study

2

แผนภาพบล็อกของนาฬิกาจับเวลา



08-Case Study

3

โมเดล VHDL ของ DIVIDER200Hz

```

library IEEE;
use IEEE.std_logic_1164.all;
entity DIVIDER200Hz is
  generic (fin: integer :=25000000; --Input
           fout: integer :=200; --Output frequency)
  port (CLK: in std_logic;
        Q : out std_logic );
end DIVIDER200Hz;

```

```

architecture RTL of DIVIDERHz is
  signal COUNT : integer range 0 to
    (fin/(2*fout)) ;
  signal qs : std_logic := '0';
  begin
  process (CLK)
  begin
    if CLK'event and CLK = '1' then
      if (COUNT >= (fin/(2*fout)-1)) then
        COUNT <= 0;
        qs <= not(qs);
      else
        COUNT <= COUNT +1;
      end if;
    end if;
  end process;
  Q <= qs ;
end RTL;

```

08-Case Study

4

โมเดล VHDL ของ DIVIDER200Hz

```

library IEEE;
use IEEE.std_logic_1164.all;
entity DIVIDER200Hz is
  generic (fin: integer :=25000000; --Input frequency
          fout: integer :=200);    --Output frequency
  port (CLK: in std_logic;
        Q : out std_logic );
end DIVIDER200Hz;

```

08-Case Study

5

โมเดล VHDL ของ DIVIDER200Hz

```

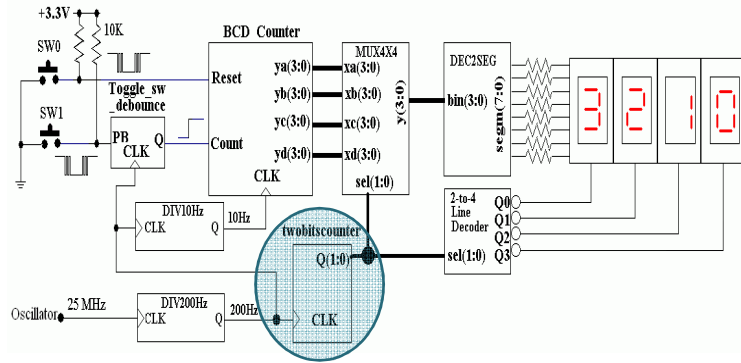
architecture RTL of DIVIDERHz is
  signal COUNT : integer range 0 to (fin/(2*fout));
  signal qs : std_logic := '0';
begin
  process (CLK)
  begin
    if CLK'event and CLK = '1' then
      if (COUNT >= (fin/(2*fout)-1)) then
        COUNT <= 0;
        qs <= not(qs);
      else
        COUNT <= COUNT +1;
      end if;
    end if;
  end process;
  Q <= qs ;
end RTL;

```

08-Case Study

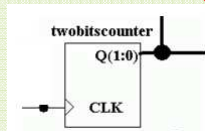
6

แผนภาพบล็อกของนาฬิกาจับเวลา



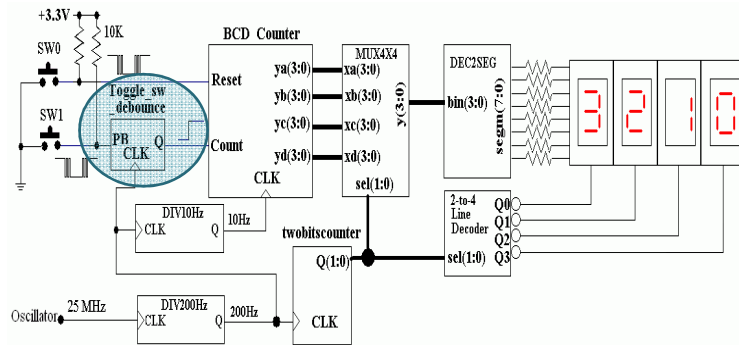
twobitscounter

```
library IEEE;
use IEEE.std_logic_1164.all;
entity twobitscounter is
    port (CLK: in std_logic;
          Q : out integer range 0 to 3);
end twobitscounter;
```



```
architecture RTL of twobitscounter is
    signal COUNT : integer range 0 to 3;
begin
    process (CLK)
    begin
        if CLK'event and CLK = '1' then
            if (COUNT >= 3) then
                COUNT <= 0;
            else
                COUNT <= COUNT + 1;
            end if;
        end if;
    end process;
    Q <= count ;
end RTL;
```

แผนภาพบล็อกของนาฬิกาจับเวลา



08-Case Study

9

Toggle_sw_debounce

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.all;
USE IEEE.STD_LOGIC_ARITH.all;
USE IEEE.STD_LOGIC_UNSIGNED.all;
```

```
ENTITY Toggle_sw_debounce IS
PORT(pb, clock_200Hz : IN STD_LOGIC;
      Q : OUT STD_LOGIC);
END Toggle_sw_debounce;
```

```
ARCHITECTURE Behavioral OF Toggle_sw_debounce IS
SIGNAL SHIFT_PB : STD_LOGIC_VECTOR(3 DOWNTO 0);
signal state : STD_LOGIC := '0';
signal qs : STD_LOGIC := '0';
BEGIN
-- Debounce clock should be approximately 10ms or 100Hz
PROCESS
BEGIN
WAIT UNTIL (clock_200Hz=EVENT) AND (clock_200Hz = '1');
-- Use a shift register to filter switch contact bounce
SHIFT_PB(2 DOWNTO 0) <= SHIFT_PB(3 DOWNTO 1);
SHIFT_PB(3) <= pb;
case shift_pb is
when "0000" => state <= '0';
when "1111" => state <= '1';
when others => state <= state;
end case;
END PROCESS;
PROCESS(state)
BEGIN
if state'event and state = '1' then
qs <= not(qs);
else
qs <= qs;
end if;
Q <= qs;
END PROCESS;
END Behavioral;
```

08-Case Study

10

Toggle_sw_debounce

ARCHITECTURE Behavioral OF Toggle_sw_debounce IS

```

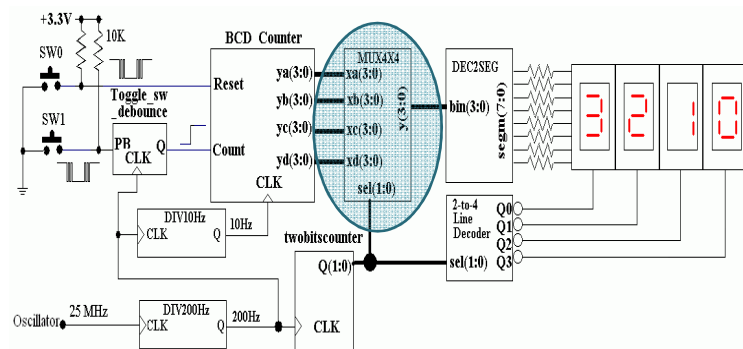
SIGNAL SHIFT_PB : STD_LOGIC_VECTOR(3 DOWNTO 0);
signal state : STD_LOGIC := '0';
signal qs : STD_LOGIC := '0';
BEGIN
  -- Debounce clock should be approximately 10ms or 100Hz
  PROCESS
  BEGIN
    WAIT UNTIL (clock_200Hz EVENT) AND (clock_200Hz = '1');
    -- Use a shift register to filter switch contact bounce
    SHIFT_PB(2 DOWNTO 0) <= SHIFT_PB(3 DOWNTO 1);
    SHIFT_PB(3) <= PB;
    case shift_pb is
      when "0000" => state <= '0';
      when "1111" => state <= '1';
      when others => state <= state;
    end case;
  END PROCESS;
  PROCESS(state)
  BEGIN
    if state'event and state = '1' then
      qs <= not(qs);
    else
      qs <= qs;
    end if;
    Q <= qs;
  END PROCESS;
END Behavioral;

```

08-Case Study

11

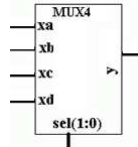
แผนภาพบล็อกของนาฬิกาจับเวลา



08-Case Study

12

MUX4X4



```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity mux4x4 is
```

```
    Port (xa, xb,xc ,xd : in integer range 0 to 15;
```

```
        sel : in integer range 0 to 3;
```

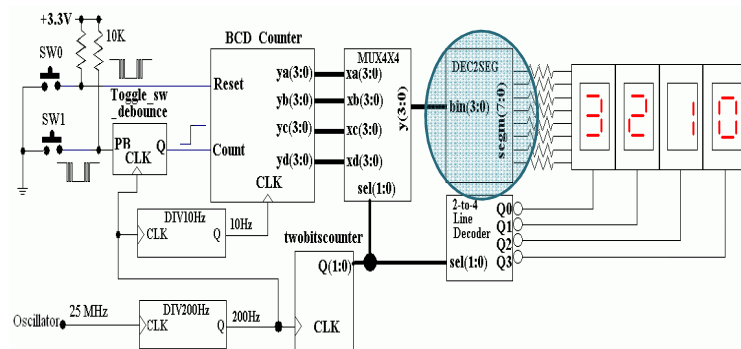
```
        y : out integer range 0 to 15);
```

```
end mux4x4;
```

```
architecture Behavioral of mux4x4 is
begin
```

```
    with sel select
        y <= xa when 0,
            xb when 1,
            xc when 2,
            xd when others;
end Behavioral;
```

แผนภาพบล็อกของนาฬิกาจับเวลา



DEC2SEG

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

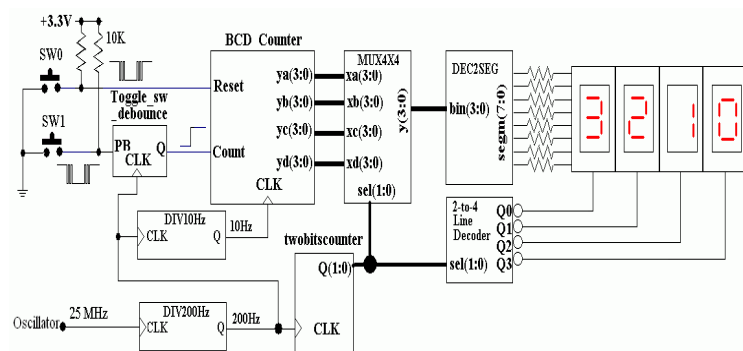
entity dec2seg is
  port (bin : in integer range 0 to 15 ;
        segm : out std_logic_vector(7 downto 0));
end dec2seg;
architecture Behavioral of dec2seg is
begin
  with bin select
    segm <= "11111100" when 0,
            "01100000" when 1,
            "11011010" when 2,
            "11110010" when 3,
            "01100110" when 4,
            "10110110" when 5,
            "10111110" when 6,
            "11100000" when 7,
            "11111110" when 8,
            "11110110" when 9,
            "00000000" when others; --Blank
end Behavioral;

```

08-Case Study

15

แผนภาพบล็อกของนาฬิกาจับเวลา



08-Case Study

16

โมเดล VHDL ของวงจรจับเวลา

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity timer is
    Port (clk, reset, count : in std_logic;
          seg : out std_logic_vector(7 downto 0);
          dig : out std_logic_vector(3 downto 0));
end timer;

architecture Behavioral of timer is
    component dec2seg is
        port (bin : in integer range 0 to 15;
              segm : out std_logic_vector(7 downto 0));
    end component;

    component twobitscounter is
        port (CLK: in std_logic;
              Q : out integer range 0 to 3);
    end component;

    component mux4x4 is
        Port (xa,xb,xc ,xd : in integer range 0 to 15;
              sel : in integer range 0 to 3;
              y : out integer range 0 to 15);
    end component;

    component DIVIDER10Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;
    component DIVIDER200Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;

    component Toggle_sw_debounce IS
        PORT(pb, clock_200Hz : IN
              Q : OUT STD_LOGIC);
    end component;

    signal ya,yb,yc, yd, bus4 : INTEGER range 0 to 15;
    signal ssel : integer range 0 to 3;
    signal clk1, clk2, count_s : std_logic;
begin
    process (CLK1, reset, count_s)
        -- BCD
        Counter
        begin
            if Reset='0' then
                ya <= 0;
                yb <= 0;
                yc <= 0;
                yd <= 0;
            else
                if count_s = '1' then
                    if CLK1='1' and CLK1'event then
                        if ya >= 9 then
                            ya <= 0;
                            if yb >= 9 then
                                yb <= 0;
                            end if;
                            if yc >= 9 then
                                yc <= 0;
                            end if;
                            if yd >= 9 then
                                yd <= 0;
                            end if;
                            yd <= yd + 1;
                        end if;
                        yc <= yc + 1;
                    end if;
                    yb <= yb + 1;
                end if;
                ya <= ya + 1;
            end if;
        end process;
    with ssel select
        dig <= "1110" when 0,
              "1101" when 1,
              "1011" when 2,
              "0111" when others;

    c1: mux4x4 port map(ya, yb, yc, yd, ssel, bus4);
    c2: dec2seg port map(bus4, seg);
    c3: twobitscounter port map(count_s, ssel);
    c4: DIVIDER200Hz port map(clk,clk2);
    c5: DIVIDER10Hz port map(clk2, clk1);
    c6: Toggle_sw_debounce port map(count_s, clk2, count_s);
end Behavioral;
    
```

โมเดล VHDL ของวงจรจับเวลา

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity timer is
    Port (clk, reset, count : in std_logic;
          seg : out std_logic_vector(7 downto 0);
          dig : out std_logic_vector(3 downto 0));
end timer;

architecture Behavioral of timer is
    component dec2seg is
        port (bin : in integer range 0 to 15;
              segm : out std_logic_vector(7 downto 0));
    end component;

    component twobitscounter is
        port (CLK: in std_logic;
              Q : out integer range 0 to 3);
    end component;

    component mux4x4 is
        Port (xa,xb,xc ,xd : in integer range 0 to 15;
              sel : in integer range 0 to 3;
              y : out integer range 0 to 15);
    end component;

    component DIVIDER10Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;
    component DIVIDER200Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;

    component Toggle_sw_debounce IS
        PORT(pb, clock_200Hz : IN
              Q : OUT STD_LOGIC);
    end component;

    signal ya,yb,yc, yd, bus4 : INTEGER range 0 to 15;
    signal ssel : integer range 0 to 3;
    signal clk1, clk2, count_s : std_logic;
begin
    component mux4x4 is
        Port (xa,xb,xc ,xd : in integer range 0 to 15;
              sel : in integer range 0 to 3;
              y : out integer range 0 to 15);
    end component;

    component DIVIDER10Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;
    component DIVIDER200Hz is
        port (CLK: in std_logic;
              Q : out std_logic );
    end component;

    component Toggle_sw_debounce IS
        PORT(pb, clock_200Hz : IN
              Q : OUT STD_LOGIC);
    end component;

    signal ya,yb,yc, yd, bus4 : INTEGER range 0 to 15;
    signal ssel : integer range 0 to 3;
    signal clk1, clk2, count_s : std_logic;
end Behavioral;
    
```

โมเดล VHDL ของวงจรถับเวลา

```

begin
process (CLK1, reset, count_s) -- BCD Counter
begin
if Reset='0' then
ya <= 0;
yb <= 0;
yc <= 0;
yd <= 0;
else
if count_s = '1' then
if CLK1='1' and CLK1'event then
if ya >= 9 then
ya <= 0;
if yb >= 9 then
yb <= 0;
if yc >= 9 then
yc <= 0;
if yd >= 9 then
yd <= 0;
else
yd <= yd + 1;
end if;
else
yc <= yc + 1;
end if;
else
yb <= yb + 1;
end if;
end if;
end if;
end process;

```

```

ya <= ya + 1;
end if;
else
ya <= ya;
yb <= yb;
yc <= yc;
yd <= yd;
end if;
end if;
end process;

with ssel select
decoder
dig <= "1110" when 0,
"1101" when 1,
"1011" when 2,
"0111" when others;

c1: mux4x4 port map(ya, yb, yc, yd, ssel, bus4);
c2: dec2seg port map(bus4, seg);
c3: twobitscounter port map(clk2, ssel);
c4: DIVIDER200Hz port map(clk, clk2);
c5: DIVIDER10Hz port map(clk2, clk1);
c6: Toggle_sw_debounce port map(count, clk2, count_s);
end Behavioral;

```

รายละเอียดหนังสือ

2. องค์ประกอบของภาษาวีเอชดีแอล
 - 2.1 ความรู้เบื้องต้นเกี่ยวกับภาษา VHDL
 - 2.2 โครงสร้างของภาษา VHDL
 - 2.3 ไบบลิอารี (Libraries)
 - 2.4 คอมโปเนนต์ (Component)
 - 2.5 การระบุจำนวน อักขระและสตริง
 - 2.6 ออบเจกต์ประเภทค่าคงที่ สัญญาณและตัวแปร
 - 2.7 ประเภทข้อมูล
 - 2.8 ข้อมูลประเภทมาตรฐาน
 - 2.9 ข้อมูลประเภทเวลา
 - 2.10 ข้อมูลประเภทอะเรย์
 - 2.11 ข้อมูลแบบแจกแจง (Enumeration Types)
 - 2.12 ข้อมูลแบบ IEEE Standard Logic Type
 - 2.13 Records
 - 2.14 Subtypes
 - 2.15 Aliases

รายละเอียดหนังสือ

3. ตัวดำเนินการของภาษา VHDL (VHDL Operators)
 - 3.1 ตัวดำเนินการและลำดับการทำงาน
 - 3.2 ตัวดำเนินการแบบลอจิก
 - 3.3 ตัวดำเนินการสำหรับการเลื่อน (Shift Operators)
 - 3.4 ตัวดำเนินการเปรียบเทียบ (Relational Operators)
 - 3.5 ตัวดำเนินการทางคณิตศาสตร์ (Arithmetic Operators)
 - 3.6 คำสั่งแบบขนาน (Concurrent Statement)
 - 3.7 คำสั่งแบบลำดับ (Sequential Statement)
 - 3.8 การใช้คำสั่ง Loop
 - 3.9 คำสั่ง WAIT Statement

รายละเอียดหนังสือ

4. โปรแกรมย่อย (Subprogram)
 - 4.1 ฟังก์ชัน (Function)
 - 4.2 โพรซีเจอร์ (Procedure)
 - 4.3 การแปลงประเภทข้อมูล (Type Conversion)
5. Testbenches
6. อุปกรณ์ลอจิกแบบโปรแกรมได้ (Programmable Logic Device)
7. การออกแบบวงจรดิจิทัลด้วย VHDL
 - 7.1 บทนำ
 - 7.2 การออกแบบวงจรคอมไบเนชันนอลลอจิกด้วย VHDL
 - 7.3 การออกแบบวงจรซีควนเชียลด้วย VHDL
 - 7.4 หน่วยความจำ